

# Proyecto Individual 1: Calculadora Full Stack

---

## Descripción del Proyecto

Este proyecto consiste en el desarrollo de una calculadora web utilizando Java 8 y Spring Boot en el backend y React en el frontend. La aplicación permite realizar operaciones básicas (suma, resta, multiplicación y división) y mantener un historial de las últimas 10 operaciones.

## Tecnologías utilizadas:

- Backend:
  - Java 8
  - Spring Boot 2.6.x
  - Spring Web
- Frontend:
  - React 18.x
  - Axios (para las peticiones HTTP al backend)
  - CSS para el diseño responsivo

## Requisitos Previos

Antes de desplegar la aplicación, asegúrate de tener instaladas las siguientes herramientas:

- Java 8 o superior
- Maven (para gestionar dependencias de Java)
- Node.js y npm (para gestionar dependencias del frontend)
- Git (opcional, para clonar el repositorio)

## Instalación

### Backend (Spring Boot con Java 8)

1. Clonar el repositorio:

```
```bash
git clone https://github.com/tuusuario/proyecto-calculadora.git
cd proyecto-calculadora/backend
```
```

## 2. Compilar el proyecto:

```
```bash
mvn clean install
```
```

## 3. Ejecutar el backend:

```
```bash
mvn spring-boot:run
```
```

Esto levantará el backend en el puerto 8080 por defecto. La API REST estará disponible en <http://localhost:8080>.

### Endpoints del Backend

- POST /api/operations: Para realizar una operación aritmética. Envía una solicitud JSON con el formato:

```
```json
{
  "operation": "suma",
  "operand1": 5,
  "operand2": 3
}
```
```

Respuesta:

```
```json
{
  "result": 8
}
```
```

- GET /api/operations/history: Devuelve el historial de las últimas 10 operaciones realizadas.

### Frontend (React)

#### 1. Instalar dependencias del frontend:

```
```bash
npm install
```
```

#### 2. Configurar el frontend:

Asegúrate de que el archivo `src/config.js` contiene la URL correcta del backend (por ejemplo, `http://localhost:8080`).

#### 3. Ejecutar el frontend:

```
```bash
npm start
```
```

...

Esto levantará la aplicación frontend en `http://localhost:3000`.

## Pruebas

### 1. Probar la funcionalidad básica:

- Abre el navegador y ve a `http://localhost:3000`.
- Realiza operaciones aritméticas desde la interfaz y observa que los resultados se muestran en tiempo real.
- Verifica que las divisiones por cero o entradas inválidas son manejadas correctamente.

### 2. Verificar el historial:

- Realiza múltiples operaciones y luego revisa la sección de historial para asegurarte de que se muestran las últimas 10 operaciones.

## Despliegue

### Desplegar el backend en producción

#### 1. Compilar la aplicación:

```
```bash
mvn clean package
```
```

Esto generará un archivo JAR en el directorio `target/`.

#### 2. Ejecutar el archivo JAR:

```
```bash
java -jar target/calculadora-0.0.1-SNAPSHOT.jar
```
```

### Desplegar el frontend en producción

#### 1. Compilar el frontend:

```
```bash
npm run build
```
```

Esto generará una carpeta `build` con todos los archivos optimizados para ser servidos en un entorno de producción.

#### 2. Desplegar el frontend:

Puedes servir los archivos generados desde un servidor estático como Nginx, Apache o utilizar plataformas como Netlify o Vercel.

## Consideraciones Finales

- Manejo de errores: Asegúrate de que las excepciones, como la división por cero, se manejan correctamente tanto en el backend como en el frontend..

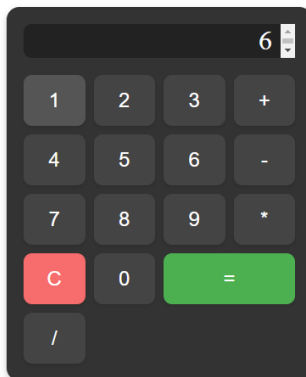
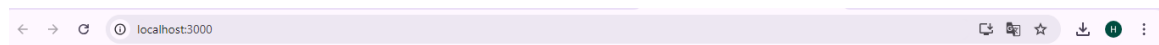
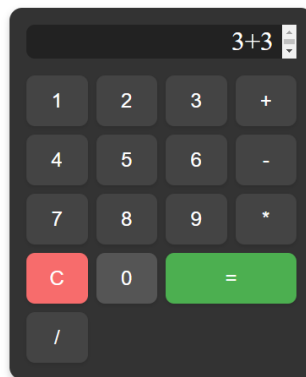
## Créditos

Desarrollador: Hugo Ismael Taque Franco

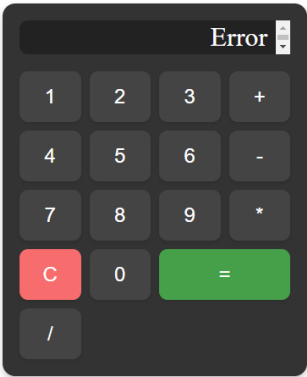
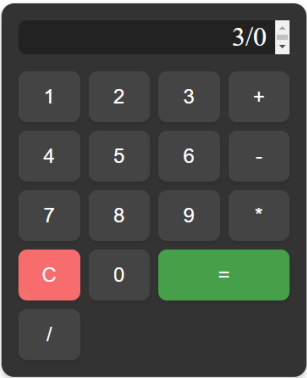
Universidad Mariano Gálvez - Curso de Programación 2

Imágenes del funcionamiento:

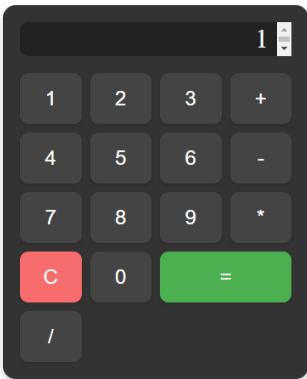
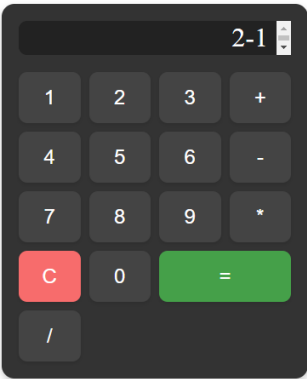
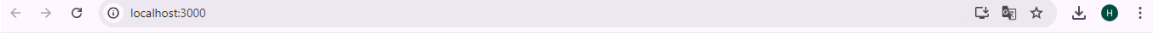
---



División por 0:



Resta:



Multiplicación:

